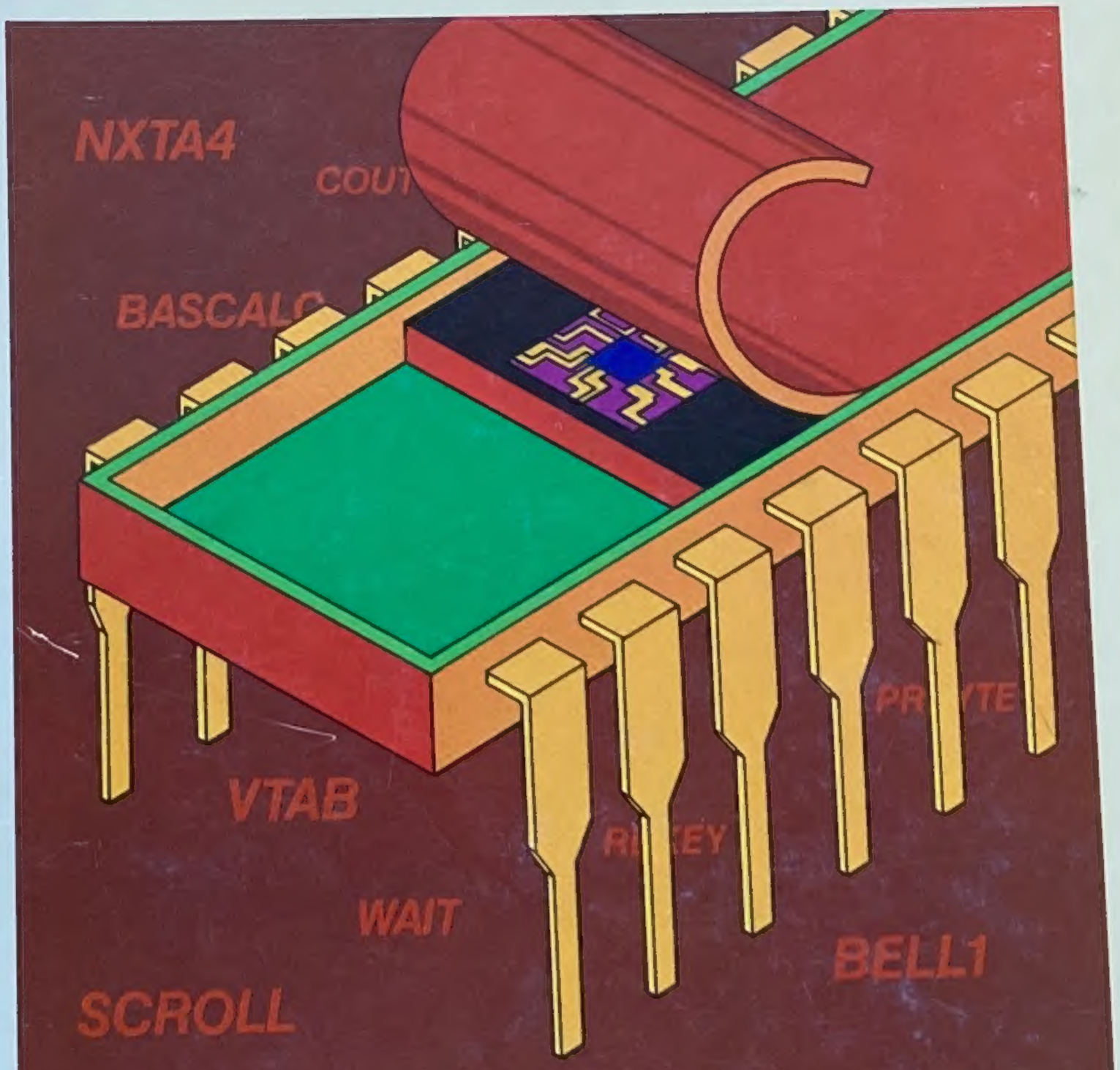
Apple II

Apple II Monitors Peeled



DISCLAIMER OF ALL WARRANTIES AND LIABILITY

APPLE COMPUTER INC. MAKES NO WARRANTIES, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS MANUAL OR WITH RESPECT TO THE SOFTWARE DESCRIBED IN THIS MANUAL, ITS QUALITY, PERFORMANCE, MERCHANTABILITY, OR FITNESS FOR ANY PARTICULAR PURPOSE. APPLE COMPUTER INC. SOFTWARE IS SOLD OR LICENSED "AS IS". THE ENTIRE RISK AS TO ITS QUALITY AND PERFORMANCE IS WITH THE BUYER. SHOULD THE PROGRAMS PROVE DEFECTIVE FOLLOWING THEIR PURCHASE, THE BUYER (AND NOT APPLE COMPUTER INC., ITS DISTRIBUTOR, OR ITS RETAILER) ASSUMES THE ENTIRE COST OF ALL NECESSARY SERVICING, REPAIR, OR CORRECTION AND ANY INCIDENTAL OR CONSEQUENTIAL DAMAGES. IN NO EVENT WILL APPLE COMPUTER INC. BE LIABLE FOR DIRECT, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT IN THE SOFTWARE, EVEN IF APPLE COMPUTER INC. HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF IMPLIED WARRANTIES OR LIABILITY FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE LIMITATION OR EXCLUSION MAY NOT APPLY TO YOU.

1Ø26Ø Bandley Drive
Cupertino, California 95Ø14
(4Ø8) 996-1Ø1Ø

APPLE Product # D2L0013
950-0018

Printed in USA

Apple II

Apple II Monitors Peeled

Function	Hex Addr	+Dec Addr	-Dec Addr	Monitor Label	Registers Destroyed
Load $DC (\) into A-reg. Backward slash indicates line input cancelled.	FD62	64866	-67Ø	CANCEL	A,X,Y
Call COUT to print A-reg. Then fall into GETLNZ.	FD64	64868	-668		
Print carriage return thru COUT.	& FD67	64871	-665	GETLNZ	A,X,Y
Load PROMPT into A-reg.	& FD6A	64874	-662	GETLN	A,X,Y
Call COUT to print A-reg.	& FD6C	64876	-66Ø		
Load X-reg with $Ø1 for passage thru backspace operation.	FD6F	64879	-657		A,X
If X=Ø goto GETLNZ to start over. Else, decrement X-reg and fall into NXTCHAR.	FD71	64881	-655	BCKSPC	A,X,Y
Call RDCHAR to get next character. If character received is ctrl-U ($95, right arrow) pick up the screen character from (BASL),Y to replace it in the A-reg.	FD75	64885	-651	NXTCHAR	A
If A-reg greater than $DF, then AND against $DF to make it upper case.	& FD7E	64894	-642	CAPTST	?A
Store A-reg to input area at IN,X. Compare to carriage return. Goto NOTCR (above) if not. Else, call CLREOL to clear the rest of the line, then print carriage return thru COUT, using RTS from that function to accomplish return to caller of keyboard input.	& FD84	649ØØ	-636	ADDINP	

IN	=$Ø2ØØ, keyboard input area.
INVFLG	is at $32 (5Ø).

OVERVIEW—TEXT OUTPUT TO THE SCREEN

The highest level of support in the Monitor for text output to the screen is scroll device support. In addition, the Monitor contains many components which support use of the screen in a formatted manner. Because there are so many ways to write text to the screen, the topic of screen output has been divided into the following sections:

TEXT OUTPUT WITHIN THE SCROLL WINDOW
describes the normal manner of text output, defining the fields in page zero which are used to control this function, and which are used in the descriptions in the following sections.

SCREEN FORMAT CONTROL
identifies the entry points by means of which display operation (full text, full graphics, mixed LORES graphics and text), Scroll Window setup, and character display mode (black on white or white on black or blinking) are established or modified.

SCROLL WINDOW DATA MANIPULATIONS
describes Monitor calls which clear all or part of the Scroll Window, set parts of the window to some user specified value, or cause conditional or unconditional scrolling of the window.

CURSOR POSITION CONTROL
describes the ways and means of moving the cursor relative to its current position, or moving it to some location independent of its current position.

GENERAL TEXT TO THE SCREEN
describes the Monitor entry points to output user program generated data to the screen or to the current output device if CSWL has been modified. Also, entry points are described to transmit standard types of output (blanks, bell code, carriage return) to the output device (generally screen).

TEXT OUTPUT WITHOUT THE SCROLL WINDOW
describes the entry points used for placing characters on the screen outside of the Scroll Window, and for reading the keyboard when echo to the Scroll Window is to be performed.

SECONDARY DISPLAY AREAS
describes various ways of using the Secondary Text area, even for limited Scroll Window functions such as allowing keyboard input echo to go to the Secondary area.

Any entry point which fits into more than one category will be found in each appropriate address table.

OUTPUT WITHIN THE SCROLL WINDOW

Scroll Window operation is compatible with printer or typewriter output in that new characters are displayed to the right of previous output, and new lines are displayed below previous lines. It is this mode of operation which is described in this section. That is, this section describes "printing" information by means of the CSWL vector to the screen or to a printer type device. The section on General Text to the Screen describes use of the screen, bypassing the CSWL vector and making direct use of the Scroll Window output routines.

The normal method provided in the Apple II for displaying output information is by "calling" COUT with the character in the A-reg for each displayable character or format control character (such as a carriage return). At COUT, a JuMP Indirect is done via the CSWL vector to the routine which will place the character on the selected medium

REGISTERS FOR BASIC MONITOR CALLS

Many of the entry points specified in this book require presetting of registers for proper operation. Following is a sample program, written for APPLESOFT, which uses Monitor calls for conversion from decimal to hex.

The theory behind the operation is that on a Monitor G command, the registers are loaded from the SAVE area before going to the location specified in PCL,H. Thus, by poking destination address into PCL,H and the required register contents into XREG, YREG, an entry point in the Monitor Go command processor can be used to pass the registers to a selected routine.

DECIMAL TO HEX CONVERSION

APPLESOFT SAMPLE PROGRAM

```
1Ø   REM CONVERT DECIMAL INPUT TO HEX OUTPUT
1ØØ  INPUT "ENTER NUMBER ";A     Read the input.
11Ø  IF A=99999 THEN END         Provide a way to end the program.
15Ø  C% = A / 256                Isolate the high byte.
2ØØ  POKE 71,C%                  Set YREG for PRNTYX call.
3ØØ  B% = A / 256                Get remainder from A/256.
31Ø  B = B% * 256                For low byte (XREG) POKE.
32Ø  B% = A - B
35Ø  POKE 7Ø,B%
4ØØ  POKE 59,249                 Set PCH to $F9.
5ØØ  POKE 58,64                  Set PCL to $4Ø.
55Ø  PRINT                       Print a blank line.
6ØØ  CALL 652Ø9                  Entry point in GO processor is FEB9.
65Ø  PRINT                       Print a blank line.
7ØØ  GOTO 1ØØ                    Go around for another number.
```

STEP AND TRACE PECULIARITIES

The Step and Trace functions in the Old Monitor incorrectly display register contents under some circumstances. The STEP routine detects and gives special attention to JSR, RTS, JMP, JMP indirect, RTI, and BRK instructions. In each case, the register contents are displayed from the SAVE area at $45-49. However, there is no SAVE call after "execution" of these instructions, as there is for normally traced instructions, so the registers displayed are those present in the SAVE area before execution of this instruction.

Therefore, on JSR and RTS, the displayed contents of the S-reg are incorrect. On the first instruction after a JSR or RTS, the S-reg displays correctly, unless that also is an RTS or JSR.

The Step and Trace routines are not incorrect in handling of a BRK instruction. That is, the address displayed for the BRK is correct, instead of being off by two bytes, because the BRK is detected by the STEP routine instead of being executed by the 65Ø2.

Although step and trace can be very helpful for some program debugging tasks, they cannot be used in tracing calls to the Monitor (generally including "print" output) or for programs which use A1L,H thru A4L,H.

Because of the lack of "CLD" at PCADJ ($F953), incorrect addresses will be displayed if you set decimal mode (SED) within the program being traced or stepped.

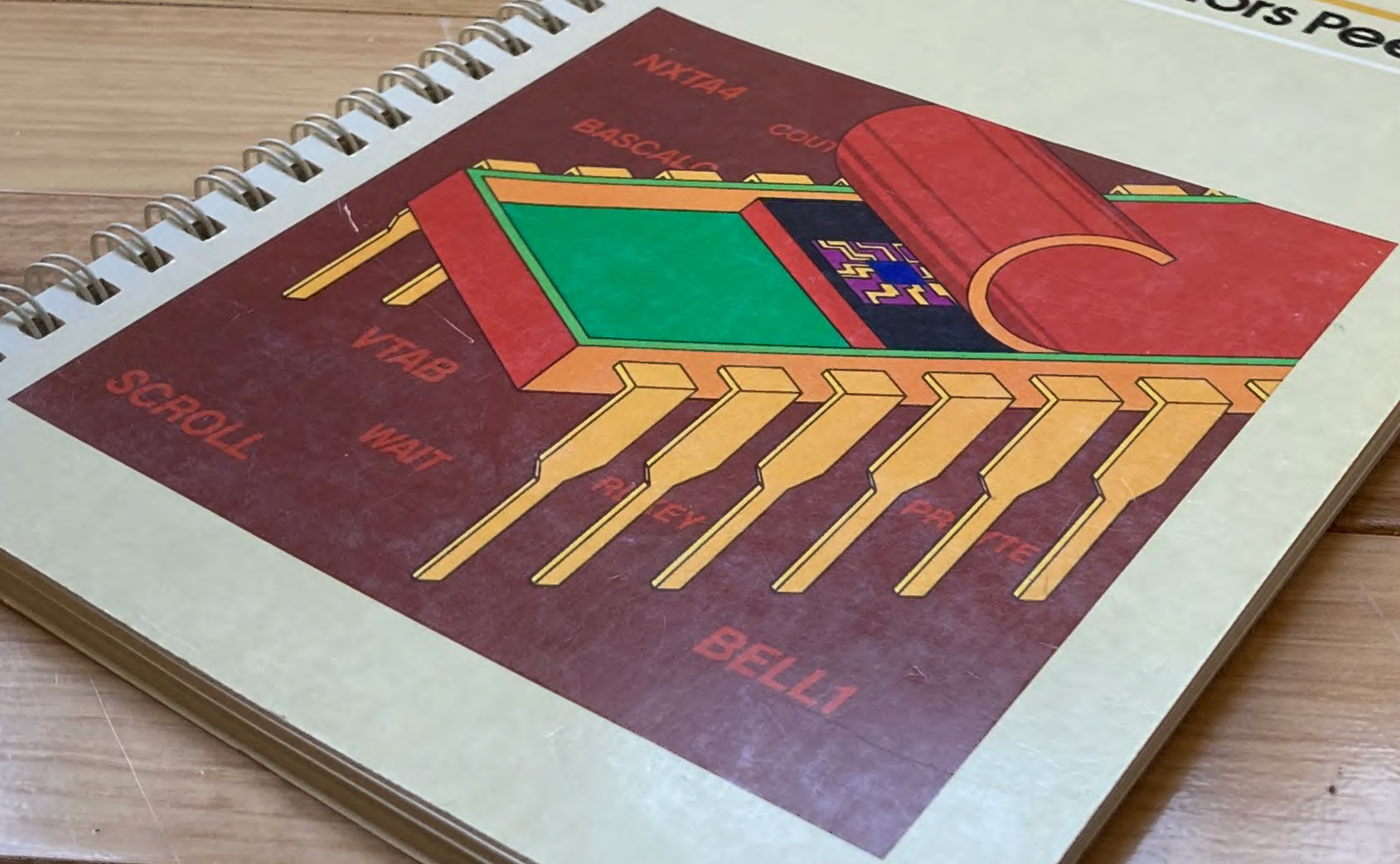
apple computer inc.

10260 Bandley Drive
Cupertino, California 95014
(408) 996-1010

950-0018-A

Apple II

Apple II Monitors Peeled



Apple II

Apple II Monitors Peeled


apple computer inc.

apple computer inc.

10260 Bandley Drive
Cupertino, California 95014
(408) 996-1010

950-0018-A